REMARKS

No claims have been amended, added or cancelled. Claims 1-68 remain pending in the application. Reconsideration is respectfully requested in light of the following remarks.

Section 103(a) Rejections:

The Examiner rejected claims 1-5, 11-17 and 22-28, 34, 35, 37, 41-45, 48-52, 54, 59, 63, 64, 67 and 68 under 35 U.S.C. § 103(a) as being unpatentable over Bittinger et al. (U.S. Patent 6,453,362) (hereinafter "Bittinger") in view of Winer ("XML-RPC for Newbies").

Regarding claim 1, contrary to the Examiner's assertion, Bittinger in view of Winer fails to teach or suggest a message in the data representation language including a credential for allowing the client access to a service configured to perform functions on behalf of clients in the distributed computing environment and sending the message to the service. Bittinger teaches the use of a ticket object created by the client that a server application can use to pass a server stub object to the client. The client can then use the server stub object to invoke remote methods on the server application (Bittinger, column 6, lines 63-67, column 7, lines 42-47, and column 8, lines 15-22).

The Examiner asserts (in both the rejection of claim 1 and in the Response to Arguments) that the client application address and the ticket identifier pair sent to Bittinger's authentication server represents the credential of Applicants' claim 1. However, Bittinger's use of the client application address and ticket identifier does not allow the client access to the authentication server, which the Examiner equates to the service of Applicants' claim 1. Instead, Bittinger's authentication server uses traditional user id and password to authenticate the client. Contrary to the Examiner's suggestion, the authentication server in Bittinger does not allow or disallow access by a client based on the client application address and ticket identifier. Instead, the authentication server

merely passes the client application address and ticket identifier to the launched application (Bittinger, column 7, lines 32-36; column 7, line 67 – column 8, line 8). Thus, the client application address and ticket identifier pair cannot be equated to a credential for allowing the client access to Bittinger's authentication server, as the Examiner erroneously contends. As Winer fails to mention anything about including a credential in a message, the combination of Bittinger and Winer fails to teach or suggest including a credential for allowing the client access to a service in a message in a data representation language sent to the service.

Bittinger in view of Winer also fails to teach or suggest the service examining the credential included in the message and performing a function on behalf of the client in accordance with the information representing the computer language method call included in the message if the credential is authentic. In contrast, Bittinger teaches the use of a separate server authenticating a client *prior to the launching* of the desired server application (Bittinger, column 3, lines 45-64 and column 8, lines 9-14). Bittinger fails to teach that any credential is verified by the service application that performs the function in accordance with a representation of the method call. In addition, Bittinger teaches that the application server only uses the received ticket identifier to obtain a client stub from a client repository in order to invoke an acknowledgement method of the client stub (Bittinger, column 3, lines 56-64).

The Examiner equates Bittinger's authentication server with the service of Applicants' claim 1 and also equates the client application address and ticket identifier pair with the credential of Applicants' claim 1. However, Bittinger fails to teach or suggest that his authentication server examines the client application address and ticket included in a client's request to launch an application. Instead, Bittinger teaches that the authentication server uses well-known login procedures, such as rlogin, and user id and password to authentication the client. Nowhere does Bittinger mention the authentication server examining the client application address and/or the ticket identifier before launching the requested application. The Examiner contends that by passing the client application address and the client ticket to the launched application, Bittinger's

authentication service is examining the application address and ticket to determine whether they are authentic. This is blatant hindsight speculation by the Examiner. Bittinger fails to teach that his authentication server examines or authenticates the application address and ticket. Instead, Bittinger, at the Examiner's cited passage, teaches only that the client application address and ticket identifier are *used as parameters* when launching the requested application. Specifically, Bittinger states, "[u]sing parameters, such as client application address .. and the identifier for the ticket, the authentication serer execution the client application command to start the application" (Bittinger, column 7, lines 32-36). Thus, the Examiner's contention that Bittinger examines and authenticates the client application address and ticket identifier is clearly erroneous.

Additionally, the Examiner contends that Bittinger's authentication server does not perform any operation if it determines that the client application address and ticket identifier are not authentic. However, there is no basis in the actual teachings of Bittinger for the Examiner's assertion. Firstly, as noted above, Bittinger's authentication server does not examine or authenticate the client application address and ticket identifier. Secondly, nowhere does Bittinger mention that his authentication server does not launch the requested application if the client application address and ticket identifier are not authentic. The Examiner is merely speculating in hindsight. Without some specific teaching or suggest from Bittinger regarding his authentication server not launching the requested application after examining the client application address and ticket identifier and determining that they (the client application address and ticket identifier) are not authentic, the Examiner's contention is merely improper speculation.

Winer also fails to teach or suggest anything regarding a client including a credential in a message sent to server or about the server authenticating such a credential included in a message. Winer therefore fails to overcome the above noted deficiencies of Bittinger. Thus, Bittinger in view of Winer clearly fails to teach or suggest a client generating a message in a data representation language, where the message includes information representing a computer programming language method call and also

includes a credential for allowing the client access to a service configured to perform functions on behalf of clients. Additionally, Bittinger in view of Winer does not teach or suggest the service examining a credential included in the message also that includes a representation of the computer programming language method call to determine whether or not the credential is authentic.

For at least the reasons above, the rejection of claim 1 is not supported by the prior art and removal thereof is respectfully requested. Furthermore, the rejection of claims 25, 45, 49 and 59 is unsupported by the prior art for similar reasons as discussed above.

Regarding claim 5, Bittinger in view of Winer fails to teach or suggest the client message endpoint attaching the credential to the message. The Examiner cites column 7, lines 1-5 of Bittinger and argues, "tStamp is an identifier used on all messages." The Examiner's interpretation of Bittinger is incorrect. The Examiner argues that Bittinger's tStamp is equivalent to a credential attached to a data representation language message. However, Bittinger teaches, "the identifier 'tstamp' may be used *to locate the ticket* within the [client-side registry] database" (emphasis added, Bittinger, column 7, lines 8-9, see also column 3, lines 47-64). Bittinger further describes a server using the tstamp to retrieve a ticket stub from the client-side registry (Bittinger, column 7, lines 41-43). After retrieving the ticket stub, the server invokes an acknowledgement method of the ticket stub. Thus, Bittinger's tStamp cannot be considered any sort of credential. Nor does Bittinger teach that a tstamp is used on all messages, contrary to the Examiner's assertion. Bittinger's use of a tstamp to allow a server application to locate and retrieve a ticket stub from a client-side registry clearly does not constitute a client message endpoint attaching the credential to a data representation language message.

In response to the above arguments, the Examiner asserts that Bittinger's tstamp "is used in combination with the client address by the authentication server to access the ticket and start the application" citing column 7, lines 27-40 of Bittinger. However, nowhere does Bittinger state that the authentication server uses the client application

address or ticket identifier (tstamp) to access the ticket. The Examiner's cited passage merely describes, as noted above regarding claim 1, how the authentication server passing the client application address and the ticket identifier to a launched application. The Examiner's assertion that Bittinger's authentication server accesses the client ticket using the client application address and ticket identifier is clearly incorrect. Bittinger teaches that the launched application uses the tstamp to access the client ticket stub. (Bittinger, column 7, lines 41-49).

The Examiner also states that Applicants have failed to provide any evidence that Bittinger's tstamp is not used on all messages. However, Bittinger teaches only that the client sends the tstamp, or ticket identifier, to the authentication when requesting the launching of an application. After the application launched, the launched application uses the tstamp to locate and obtain the full ticket, or tstub, and then sends the client a server stub, which the client subsequently uses to make requests to the server application. Bittinger fails to mention anywhere that the client includes the tstamp when making requests through the server stub. The Examiner also states, "Bittinger clearly discloses that the tstamp is included in all messages representing a method call", citing column 7, lines 27-40 and item 28 of FIG 3 of Bittinger. However, the cited passages do not mention that the tstamp is included in *all messages that represent a method call*, as the Examiner asserts. In fact, the Examiner's cited passage states only that Bittinger's authentication server uses the tstamp as a parameter when starting the requested application. Nowhere does Bittinger mention the client including the tstamp in all message that represent a method call. Applicants remind the Examiner that it is he, not the applicants, who shoulders the burden to prove that cited art teaches all limitations of the claims.

As Winer does not teach or suggest anything about a client message endpoint attaching credentials to messages, Winer fails to overcome the above noted deficiencies of Bittinger. Thus, the Examiner's combination of Bittinger in view of Winer does not teach or suggest the client message endpoint attaching the credential to the message. For

at least the reasons above, the rejection of claim 5 is not supported by the prior art and removal thereof is respectfully requested. Similar arguments apply to claim 28.

Regarding claim 15, the Examiner lists claim 15 both as rejected and objected to but allowable if rewritten in independent form. Since the Examiner does not provide an actual rejection of claim 15, Applicants assume the Examiner intended to only object to claim 15.

The Examiner rejected claims 21, 41 and 67 under 35 U.S.C. § 103(a) as being unpatentable over Bittinger in view of Winer and in further of the Instaweb Online Computing Dictionary (hereinafter "Instaweb"). Applicants respectfully traverse this rejection for at least the reasons presented above regarding their respective independent claims.

Applicants also assert that numerous ones of the dependent claims recite further distinctions over the cited art. However, since the rejection has been shown to be unsupported for the independent claims, a further discussion of the dependent claims is not necessary at this time.

Allowed Claims:

Claims 55-58 have been allowed by the Examiner.

Claims Objected To But Otherwise Allowable:

Claims 6-10, 15, 18-20, 29-33, 36, 38-40, 46, 47, 53, 60-62, 65 and 66 were rejected as being dependent upon a rejected base claim but otherwise allowable if rewritten in independent form. Applicants assert that claims 6-10, 15, 18-20, 29-33, 36, 38-40, 46, 47, 53, 60-62, 65 and 66 are allowable as depending from patentably distinct

base claims. Applicants therefore respectfully request allowance of claims 6-10, 15, 18-20, 29-33, 36, 38-40, 46, 47, 53, 60-62, 65 and 66 as currently pending.

CONCLUSION

Applicants submit the application is in condition for allowance, and prompt notice to that effect is respectfully requested.

If any fees are due, the Commissioner is authorized to charge said fees to Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C. Deposit Account No. 501505/5181-67300/RCK.

Also enclosed herewith are the following items:

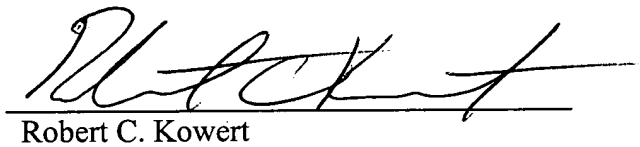
☒ Return Receipt Postcard

☐ Petition for Extension of Time

☐ Notice of Change of Address

☐ Other:

Respectfully submitted,



Robert C. Kowert
Reg. No. 39,255
ATTORNEY FOR APPLICANT(S)

Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C.
P.O. Box 398
Austin, TX 78767-0398
Phone: (512) 853-8850

Date: November 16, 2005